



## Information Technology

### Sample Analysis at Mars Instrument Simulator

**A novel tool will optimize and validate science operation plans of the instrument on the surface of Mars with a turnaround time of a few hours instead of multiple days.**

*Goddard Space Flight Center, Greenbelt, Maryland*

The Sample Analysis at Mars Instrument Simulator (SAMSIM) is a numerical model dedicated to plan and validate operations of the Sample Analysis at Mars (SAM) instrument on the surface of Mars. The SAM instrument suite, currently operating on the Mars Science Laboratory (MSL), is an analytical laboratory designed to investigate the chemical and isotopic composition of the atmosphere and volatiles extracted from solid samples. SAMSIM was developed using Matlab and Simulink libraries of MathWorks Inc. to provide MSL mission planners with accurate predictions of the instrument electrical, thermal, mechanical, and fluid responses to scripted commands. This tool is a first example of a multi-purpose, full-scale numerical modeling of a flight instrument with the purpose of supplementing or even eliminating entirely the need for a hardware engineer model during instrument development and operation.

SAMSIM simulates the complex interactions that occur between the instrument Command and Data Handling unit (C&DH) and all subsystems during the execution of experiment sequences. A typical SAM experiment takes many hours to complete and involves hundreds of components. During the simulation, the electrical, mechanical, thermal, and gas dynamics states of each hardware component are accurately

modeled and propagated within the simulation environment at faster than real time. This allows the simulation, in just a few minutes, of experiment sequences that takes many hours to execute on the real instrument.

The SAMSIM model is divided into five distinct but interacting modules: software, mechanical, thermal, gas flow, and electrical modules. The software module simulates the instrument C&DH by executing a customized version of the instrument flight software in a Matlab environment. The inputs and outputs to this synthetic C&DH are mapped to virtual sensors and command lines that mimic in their structure and connectivity the layout of the instrument harnesses. This module executes, and thus validates, complex command scripts prior to their up-linking to the SAM instrument. As an output, this module generates synthetic data and message logs at a rate that is similar to the actual instrument.

The mechanical module simulates the actions of a number of mechanical components of the SAM instrument. This module contains the valves, the pumps, and the Sample Manipulation System (SMS). The response of each of these elements is modeled to produce outputs of state variables (velocity, position, force, etc.) that duplicate the behavior of the real elements.

The thermal module simulates the action of ≈60 heaters and their local thermal impact on the instrument. It also predicts the individual temperatures sensed by ≈70 thermistors and transmitted to the C&DH.

The gas flow module simulates the gas dynamics (time-dependent pressures and gas flows) in the Gas-Processing System (GPS) of SAM. The conductance of each gas transfer element (pipe, manifold, leak, pump, etc.) is dynamically computed as a function of the component dimension, gas composition, pressure, and temperature.

The electric module captures the electrical behavior of motors, control boards, Bayard-Alpert gauges, detectors, valves, heaters, and all components that contribute to the instrument power profile.

The behavior of each element of the SAMSIM model is reconstructed by accounting for all the relevant modules to that element. Complex components like pumps integrate contributions from modules to simulate the evolution of their command logic, rotor speeds, currents, temperatures, and throughputs.

*This development was conducted by Mehdi Benna of the Center for Research and Exploration in Space Science & Technology (CREST) at Goddard Space Flight Center and Tom Nolan of Nolan Engineering, LLC. Further information is contained in a TSP (see page 1). GSC-16566-1*

### Access Control of Web- and Java-Based Applications

*NASA's Jet Propulsion Laboratory, Pasadena, California*

Cybersecurity has become a great concern as threats of service interruption, unauthorized access, stealing and altering of information, and spreading of viruses have become more prevalent and serious.

Application layer access control of applications is a critical component in the overall security solution that also includes encryption, firewalls, virtual pri-

vate networks, antivirus, and intrusion detection. An access control solution, based on an open-source access manager augmented with custom software components, was developed to provide protection to both Web-based and Java-based client and server applications.

The DISA Security Service (DISA-SS) provides common access control capabilities for AMMOS software applica-

tions through a set of application programming interfaces (APIs) and network-accessible security services for authentication, single sign-on, authorization checking, and authorization policy management.

The OpenAM access management technology designed for Web applications can be extended to meet the needs of Java thick clients and stand-

alone servers that are commonly used in the JPL AMMOS environment. The DISA-SS reusable components have greatly reduced the effort for each AMMOS subsystem to develop its own access control strategy.

The novelty of this work is that it leverages an open-source access management product that was designed for Web-

based applications to provide access control for Java thick clients and Java standalone servers. Thick clients and standalone servers are still commonly used in businesses and government, especially for applications that require rich graphical user interfaces and high-performance visualization that cannot be met by thin clients running on Web browsers.

*This work was done by Kam S. Tso and Michael J. Pajewski of Caltech for NASA's Jet Propulsion Laboratory. For more information, contact [iaoffice@jpl.nasa.gov](mailto:iaoffice@jpl.nasa.gov).*

*The software used in this innovation is available for commercial licensing. Please contact Dan Broderick at [Daniel.F.Broderick@jpl.nasa.gov](mailto:Daniel.F.Broderick@jpl.nasa.gov). Refer to NPO-48435.*

---

## Tool for Automated Retrieval of Generic Event Tracks (TARGET)

**A generalized algorithm implementation is applied to scientific data sets for establishing events, such as tornadoes, both spatially and temporally.**

*Goddard Space Flight Center, Greenbelt, Maryland*

Methods have been developed to identify and track tornado-producing mesoscale convective systems (MCSs) automatically over the continental United States, in order to facilitate systematic studies of these powerful and often destructive events. Several data sources were combined to ensure event identification accuracy. Records of watches and warnings issued by National Weather Service (NWS), and tornado locations and tracks from the Tornado History Project (THP) were used to locate MCSs in high-resolution precipitation observations and GOES infrared (11-micron) Rapid Scan Operation (RSO) imagery. Thresholds are then applied to the latter two data sets to define MCS events and track their developments.

MCSs produce a broad range of severe convective weather events that are significantly affecting the living conditions of the populations exposed to them. Understanding how MCSs grow and develop could help scientists improve their weather prediction models, and also provide tools to decision-makers whose goals are to protect populations and their property.

Associating storm cells across frames of remotely sensed images poses a difficult problem because storms evolve, split, and merge. Any storm-tracking method should include the following processes: storm identification, storm tracking, and quantification of storm intensity and activity.

The spatiotemporal coordinates of the tracks will enable researchers to obtain other coincident observations to conduct more thorough studies of these events. In addition to their tracked locations, their areal extents, precipitation intensities, and accumulations — all as functions of their evolutions in time — were also obtained and recorded for these events. All parameters so derived can be catalogued into a moving object database (MODB) for custom queries.

The purpose of this software is to provide a generalized, cross-platform, pluggable tool for identifying events within a set of scientific data based upon specified criteria with the possibility of storing identified events into a searchable database. The core of the application uses an implementation of the connected component labeling (CCL) algo-

rithm to identify areas of interest, then uses a set of criteria to establish spatial and temporal relationships between identified components. The CCL algorithm is used for identifying objects within images for computer vision. This application applies it to scientific data sets using arbitrary criteria.

The most novel concept was applying a generalized CCL implementation to scientific data sets for establishing events both spatially and temporally. The combination of several existing concepts (pluggable components, generalized CCL algorithm, etc.) into one application is also novel. In addition, how the system is designed, i.e., its extensibility with pluggable components, and its configurability with a simple configuration file, is innovative. This allows the system to be applied to new scenarios with ease.

*This work was done by Thomas Chune of Goddard Space Flight Center; Shawn Freeman, Carlos Cruz, and Robert Burns of Northrop Grumman; Kwo-Sen Kuo of Caelum Research Corporation; and Jules Kouatchou of TetraTech AMT. Further information is contained in a TSP (see page 1). GSC-16665-1*

---

## Bilayer Protograph Codes for Half-Duplex Relay Channels

**The proposed code is constructed by synthesizing a bilayer structure with a protograph.**

*NASA's Jet Propulsion Laboratory, Pasadena, California*

Direct to Earth return links are limited by the size and power of lander devices. A standard alternative is provided by a two-hops return link: a proximity link (from lander to orbiter relay) and a deep-space link (from orbiter relay to Earth). Although direct to Earth return

links are limited by the size and power of lander devices, using an additional link and a proposed coding for relay channels, one can obtain a more reliable signal. Although significant progress has been made in the relay coding problem, existing codes must be painstakingly op-

timized to match to a single set of channel conditions, many of them do not offer easy encoding, and most of them do not have structured design.

A high-performing LDPC (low-density parity-check) code for the relay channel addresses simultaneously two important